SYSTEM, TOOLS AND METHODS FOR VIEWING TEXTUAL DOCUMENTS, EXRACTING KNOWLEDGE THEREFROM AND CONVERTING THE KNOWLEDGE INTO OTHER FORMS OF REPRESENTATION OF THE KNOWLEDGE

FIELD OF THE INVENTION

[0001] The present invention pertains to the field of biological data management. More particularly, the present invention relates to identifying and interrelating biological information contained within textual documents with other types of data, such as biological diagrams and experimental data, and using such information interactively with biological diagrams.

BACKGROUND OF THE INVENTION

[0002] The advent of high-throughput experimental technologies for molecular biology have resulted in an explosion of data and a rapidly increasing diversity of biological measurement data types. Examples of such biological measurement types include gene expression from DNA microarray or Quantitative PCR experiments, protein identification from mass spectrometry or gel electrophoresis, cell localization information from flow cytometry, phenotype information from clinical data or knockout experiments, genotype information from association studies and DNA microarray experiments, etc. This data is rapidly changing; new technologies frequently generate new types of data. In addition to data from their own experiments, biologists also utilize a rich body of available information from Internet-based sources, e.g. genomic and proteomic databases, and from the scientific literature. The structure and content of these sources is also rapidly evolving. The software tools used by molecular biologists need to gracefully accommodate new and rapidly changing data types.

[0003] One manner in which biologists use these experimental data and other sources of information is in an effort to piece together interpretations and form hypotheses about biological processes, also referred to as building biological models. Textual documents are often relied upon as a source of "known" information, which can be used to compare to experimental data and/or in constructing biological diagrams/models, to confirm or refute hypotheses or data

resulting from experimentation for example.

[0004] A large number of systems have been developed to automatically build biological models from these various sources of biological data. However, these tools suffer from at least two major limitations: they lack accuracy in extracting knowledge for building the biological models, and also, they cannot incorporate a user's changing contexts and hence are not true to users' intents. Manual building of models has the strength that the model is true to the user's intent. By manually building, the model builder can capture all the nuances and subtleties that only a human can provide. There are significant disadvantages in manually building such models, however, in that the process of building biological models is tedious and error prone, particularly as data and models get larger and more complex.

[0005] Likewise, manual extraction of knowledge from text has the advantage that the extractions made are each individually chosen by the user and therefore only relevant data is generally extracted by this method. Again, however, this method is extremely tedious, time-consuming, and inefficient.

[0006] There are currently systems that can generate biological network information, such as protein-protein interaction networks, via knowledge extraction from text, and which display their output via network diagrams. Examples of these are Ariadne Genomics (www.ariadnegenomics.com); Apelon (www.apelon.com), BioSentients (www.biosentients.com); BioWisdom (www.biowisdom.co.uk); Cellomics CellSpace ™ (http://www.cellomics.com/products/cellspace/); Definiens (www.definiens.de); Gene Ed/Reel Two (www.geneed.com www.reeltwo.com); Incellico (www.incellico.com); Ingenuity (www.ingenuity.com); Insightful (www.insightful.com); Iridescent (http://innovation.swmed.edu./Biocomputing/Computing.htm); Pre-BIND (http://www.binddb.org); PubGene (http://www.pubgene.com/); Virtual Genetics (www.vglab.com); and XMine (http://www.x-mine.com/). These systems rely on statistical and linguistic natural language processing to automatically pre-compute protein-protein interactions from scientific text into a database. They therefore present a completely generated network to the user; there is no opportunity for the user to guide and/or improve the process of knowledge

extraction by disambiguating and/or assigning directionality or causality. These systems are also plagued by numerous inaccuracies and inconsistencies, leading to skepticism by would-be users in real practice.

[0007] In view of the existing systems, what is needed are systems methods and tools capable of not only easily and semi-automatically (i.e., providing the opportunity for user input and/or editing) extracting knowledge or relevant information from textual documents, but which also provide for user interaction to guide and improve the resultant information that is extracted, such as by error correction, disambiguation, and/or custom tailoring to the user's needs.

SUMMARY OF THE INVENTION

[0008] The present invention provides systems, methods and computer readable media for manipulating biological data. The present invention provides tools to convert free-form information in scientific text to a structured, machine readable format, such as the local format, which can then be used to link various forms of biological data for their interactive use.

[0009] The present invention provides systems, tools and methods for providing interactive capabilities for user involvement in extracting and disambiguating biological information in scientific text to be converted into a structured format, such as the local format. The local format can then be used in generating a biological diagram. For example, one such tool provides a text viewer into which at least a portion of a textual document may be imported and viewed; means for text mining the text having been imported into the text viewer; a list-based text editor that lists entities and interactions having been identified by the text mining; and means for assigning directionality to the listed interactions.

[0010] The entities and interactions listed each point back to a location of the portion of the textual document where it was identified. Slots are associated with each interaction listed so that a user can identify one or more of the listed entities involved in the interaction, and assign the roles played by each of these entities, in a particular interaction.

[0011] A canvas area may be provided for diagrammatically representing entities and interactions having been identified by text mining, or biological diagrams may be generated based on identified entities and interactions and displayed on a

separate graphical viewer. At least one pre-designed blank graphical rendering representing a particular type of interaction may be provided for use in population thereof in the canvas area. Population may be with one or more of entities and interactions identified during text mining of a textual document.

[0012] User context may be provided to process scientific text to identify entities and interactions within the textual document.

[0013] Textual documents may be processed in batch mode, including any and all of the steps of textual searching to identify relevant documents, identification and local formatting of entities and interactions within textual documents; disambiguation of interactions that have been identified, highlighting of locations of identified entities and/or interactions in the textual documents where they were extracted, construction of one or more biological diagrams using identified entities and interactions, and alias management of alias names for entities and/or interactions.

[0014] A tool for building biological networks of interactions is provided, which includes a text viewer, means for text mining, a list-based text editor, means for assigning directionality to the listed interactions or additionally converting interactions into a local format; and means for selecting interactions and associated entities in the list-based editor, merging common entities and displaying a resulting network of the interactions in the network viewer.

[0015] A tool for comparing extracted biological knowledge against an existing biological diagram is provided, including a text viewer, means for text mining, a list-based text editor, a diagram viewer and means for importing at least a portion of an existing biological diagram into the diagram viewer; means for overlaying the identified entities and interactions on the existing biological diagram that is displayed in the diagram viewer, and means for visually distinguishing the overlaid entities and interactions from a remainder of the displayed biological diagram.

[0016] By use of the present invention, a user may easily and conveniently construct diagrammatic representations of data/text that can be used to make an interactive biological diagram.

[0017] A user context is provided as a basis upon which text mining and other related functions of the present system function. The user context may be

readily edited by a user. The user context may be created by the user or a new user context can be created to replace an existing context.

[0018] Alias management functionality is provided so that functions may be run concurrently with regard to an entity (concept) and/or interaction (relationship), as well as any known existing aliases.

[0019] Batch mode processing of textual documents is also provided.

[0020] Methods for using each of the above tools and systems, either alone or in any usable combination are also provided.

[0021] These and other advantages and features of the invention will become apparent to those persons skilled in the art upon reading the details of the invention as more fully described below.

BRIEF DESCRIPTION OF THE DRAWINGS

[0022] Fig. 1 is a schematic representation of a system for facilitating interaction, comparisons, overlays, etc. of information from different categories, such as textual material (e.g., scientific literature), experimental data and biological diagrams.

[0023] Fig. 2 shows a screen shot of an example of a text viewer tool according to the present invention.

[0024] Fig. 3 is an example of a text viewer tool including a diagramming window therewith.

[0025] Figs. 4A-4B show the tool of Fig. 2 after various stages of populating a graphical representation in the canvas of the diagramming window.

[0026] Fig. 5 shows a tool according to the present invention which may be used to build networks of interactions by composing entities, interactions, and graphical renderings. A user can associate the resultant network with experimental data values, performing an informal verification of the putative network against actual data.

[0027] Figs. 6A-6B show a tool providing the ability to compare extracted knowledge against an existing biological network. Fig. 6B shows detail from a pane in Fig. 6A.

[0028] Fig. 7 is a block diagram illustrating an example of a generic computer system which may be used in implementing the present invention.

DETAILED DESCRIPTION OF THE INVENTION

[0029] Before the present systems, tools and methods are described, it is to be understood that this invention is not limited to particular software, hardware, software language or symbols described, as such may, of course, vary. It is also to be understood that the terminology used herein is for the purpose of describing particular embodiments only, and is not intended to be limiting, since the scope of the present invention will be limited only by the appended claims.

[0030] Unless defined otherwise, all technical and scientific terms used herein have the same meaning as commonly understood by one of ordinary skill in the art to which this invention belongs. Although any methods and materials similar or equivalent to those described herein can be used in the practice or testing of the present invention, the preferred methods and materials are now described. All publications mentioned herein are incorporated herein by reference to disclose and describe the methods and/or materials in connection with which the publications are cited.

[0031] It must be noted that as used herein and in the appended claims, the singular forms "a", "and", and "the" include plural referents unless the context clearly dictates otherwise. Thus, for example, reference to "a concept" includes a plurality of such concepts and reference to "the diagram" includes reference to one or more diagrams and equivalents thereof known to those skilled in the art, and so forth.

[0032] The publications discussed herein are provided solely for their disclosure prior to the filing date of the present application. Nothing herein is to be construed as an admission that the present invention is not entitled to antedate such publication by virtue of prior invention. Further, the dates of publication provided may be different from the actual publication dates which may need to be independently confirmed.

DEFINITIONS

[0033] In the present application, unless a contrary intention appears, the following terms refer to the indicated characteristics.

[0034] The term “biological diagram” or “biological model”, as used herein, refers to any graphical image, stored in ay type of format (e.g., GIF, JPG, TIFF, BMP, etc.) which contains depictions of concepts found in biology. Biological diagrams include, but are not limited to, pathway diagrams, cellular networks, signal transduction pathways, regulatory pathways, metabolic pathways, protein-protein interactions, interactions between molecules, compounds, or drugs, and the like.

[0035] A “biological concept” refers to any concept from the biological domain that can be described using one or more “nouns” according to the techniques described herein.

[0036] An “entity” or “item” is defined herein as a subject of interest that a researcher is endeavoring to learn more about, and may also be referred to as a biological concept, i.e., “entities” are a subset of “concepts”. For example, an entity or item may be one or more genes, proteins, molecules, ligands, diseases, drugs or other compounds, textual or other semantic description of the foregoing, or combinations of any or all of the foregoing, but is not limited to these specific examples.

[0037] An “interaction” as used herein, refers to some association relating two or more entities. Co-occurrence of entities in an interaction implies that there exists some relationship between those entities. Entities may play a number of roles within an interaction. The structure of roles in an interaction determines the nature of the relationship(s) amongst the various entities that fill those roles. An empty role in an interaction can be referred to as a “slot” or placeholder, where an entity may be assigned.

[0038] When one item is indicated as being "remote" from another, this is referenced that the two items are at least in different buildings, and may be at least one mile, ten miles, or at least one hundred miles apart.

[0039] "Communicating" information references transmitting the data representing that information as electrical signals over a suitable communication channel (for example, a private or public network). "Forwarding" an item refers to any means of getting that item from one location to the next, whether by physically transporting that item or otherwise (where that is possible) and includes, at least

in the case of data, physically transporting a medium carrying the data or communicating the data.

[0040] A "processor" references any hardware and/or software combination which will perform the functions required of it. For example, any processor herein may be a programmable digital microprocessor such as available in the form of a mainframe, server, or personal computer (desktop or portable). Where the processor is programmable, suitable programming can be communicated from a remote location to the processor, or previously saved in a computer program product (such as a portable or fixed computer readable storage medium, whether magnetic, optical or solid state device based). For example, a magnetic or optical disk may carry the programming, and can be read by a suitable disk reader communicating with each processor at its corresponding station.

[0041] "May" means optionally.

[0042] Methods recited herein may be carried out in any order of the recited events which is logically possible, as well as the recited order of events.

[0043] "Local format" refers to a restricted grammar/language used to represent extracted semantic information from diagrams, text, experimental data, etc., so that all of the extracted information is in the same format and may be easily exchanged and used in together. The local format can be used to link information from diverse categories, and this may be carried out automatically. The information that results in the local format can then be used as a precursor for application tools provided to compare experimental data with existing textual data and biological models, as well as with any textual data or biological models that the user may supply, for example.

[0044] A "node" as used herein, refers to an entity, which also may be referred to as a "noun" (in a local format, for example). Thus, when data is converted to a local format according to the present invention, nodes are selected as the "nouns" for the local format to build a grammar, language or Boolean logic.

[0045] A "link" as used herein, refers to a relationship or action that occurs between entities or nodes (nouns) and may also be referred to as a "verb" (in a local format, for example). Verbs are identified for use in the local format to construct a grammar, language or Boolean logic. Examples of verbs, but not limited to these, include upregulation, downregulation, inhibition, promotion,

bind, cleave and status of genes, protein-protein interactions, drug actions and reactions, etc.

[0046] Although it is currently possible to identify interactions between biological entities from textual documents, for example, using automated text mining tools, (e.g., it is possible to identify the "nouns" and "verbs" used in describing an interaction involving entities), it was not heretofore possible to unambiguously identify causality or directionality of the interactions. A method and system for knowledge extraction is described in co-pending commonly owned Application Serial No. 10/154,524 titled "System and Method for Extracting Pre-Existing Data from Multiple Formats and Representing Data in a Common Format for Making Overlays", filed on May 22, 2002. Application Serial No. 10/154,524 is hereby incorporated by reference herein, in its entirety, by reference thereto. Further, a method and system for using local user context to extract relevant knowledge is described in co-pending and commonly assigned Application Serial No. 10/155,304 filed May 22, 2002 and titled "System, Tools and Methods to Facilitate Identification and Organization of New Information Based on Context of User's Existing Information". Application Serial No. 10/155,304 is hereby incorporated by reference herein, in its entirety, by reference thereto. Described are methods and systems wherein automated text mining techniques are used to extract "nouns" (e.g. biological entities) and "verbs" (e.g. relationships) from sentences in scientific text. Thus, knowledge extraction from scientific literature, e.g. via text mining, can identify biological entities that are involved in a relationship, for example a promotion interaction involving two genes. The resulting interpretation is represented in a restricted grammar, referred to as "local format".

[0047] The present invention converts text to the local format using an interactive text viewing tool. This tool can automatically identify and extract entities and relationships found in a passage of text, and then provide an interface by which a user can interactively refine and disambiguate the extracted knowledge, which the present invention converts to a local format, thereby greatly improving the accuracy and reliability of the knowledge generated, as a result of the process. The local format serves as a structured way for the user to review and encode the relevant knowledge contained in scientific text. It also

serves as a biological object model that can be manipulated by other computational tools.

[0048] The present invention extends the functionality and versatility of the local format by augmenting automated tools to enable the user to interact with the knowledge extraction process to clarify and/or correct the results of the process by disambiguation, and hence, transform free-form text into the structured representation of the local format. The present invention sits on top of the local format infrastructure and provides an interface by which a user can create local format objects and/or modify existing local format objects. The present invention allows associations to be made between local format objects and entities, concepts, interactions and/or relationships described in textual data, and provides various interfaces which facilitate a user's manipulation of such data as well as the underlying local format objects.

[0049] Fig. 1 is a schematic representation of a system 1000 for facilitating interaction, comparisons, overlays, etc. of information from different categories, such as textual material (e.g., scientific literature), experimental data and biological diagrams. Using a local format infrastructural layer 400 (as described in co-pending Application Serial No. 10/154,524 for example, knowledge from one representation (text, data or graphical) may be transformed or linked to one or more other of the representations. This allows combining knowledge from different representations for comparison purposes, for constructing new and more detailed representations of knowledge, and the like. At the local format level 400 the knowledge is converted to a canonical or abstract representation. This abstract representation serves as a common language (local format) which can be used for textual representations, data representations and graphical representations of knowledge.

[0050] While many different textual editors or viewers may be used to access textual representations of knowledge and input such knowledge for conversion to the local format (some may also even data mine and automatically extract nouns and verbs, as noted above), textual viewer 100, according to the present invention, provides for further user interaction for improvement of the knowledge gathered, as well as improvement of the accuracy when converting such knowledge to a local format.

[0051] A diagram viewer 200 may be used to view biological diagrams, import graphical knowledge from the same and convert it to the local format at 400 for use with text and/or data. Further special features for conversion of biological diagrams, as well as construction of biological diagrams, which may be accompanied with use of the local format can be found in co-pending, commonly owned Application (Application Serial No. not yet assigned, Attorney's Docket No. 10030687-1) filed on even date herewith, and titled "Method and System for Data Overlay and Navigation on a Biological Diagram". Application (Application Serial No. not yet assigned, Attorney's Docket No. 10030687-1) is incorporated herein, in its entirety, by reference thereto.

[0052] Experimental data may be imported and converted to the local format, using a data viewer 300, for overlays on textual documents, biological diagrams, or incorporation of such knowledge with textual knowledge and/or graphical knowledge, through conversion of all types to a local format. However a specific data viewer having functionality analogous to that of the text viewer 100 according to the present invention, or to the functionality of the diagram viewer described in Application Serial No. (Application Serial No. not yet assigned, Attorney's Docket No. 10030687-1) has not yet been developed, as the complexities in addressing specific requirements for forming relationships among individual data points and disambiguating such relationships is much more challenging than the tasks presented by either textual knowledge or diagram knowledge.

[0053] Thus, the infrastructural layer 400 provides the means/data model by which knowledge from different sources may be converted and displayed at various endpoints (applications) such as text viewer 100, diagram viewer 200 and data viewer 300.

[0054] Fig. 2 shows a screen shot of an example of a tool (textual viewer) 100 according to the present invention, which allows a user to extract knowledge from textual forms of biological knowledge. Viewer 100 is configured to involve a user in the process of disambiguation. Tool 100 includes a list-based text editor that allows users to make textual assignments for the entities and interactions identified in a textual document by filling in textual slots or areas of

the viewer 100 provided for user input. In the example shown, an excerpt from a publication identified in PubMed (http:www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=PubMed) was inputted into the text viewer window 110 of tool 100, such as by automatic extraction or cutting and pasting, for example. Upon selection of the "Analyze Text" button 102, tool 100 uses the user context and local format tools and methods described in Application Serial Nos. 10/154,524 and 10/155,304 to identify the nouns and verbs contained within the textual excerpt, extract them, and list them in the entities window 120 and interactions window 130, as shown. Each item in the list structure for interactions 130 is a visual representation of the underlying local format, restricted grammar representation of that interaction.

[0055] Automated analysis using lexicons for entities and interactions are used to identify the interesting (e.g., those nouns and verbs matching those in the user context or matched by a lookup service for aliases, such as Biological Naming System) nouns and verbs in every sentence of the text. The lexicons are part of the user context provided by tool 100. The lexicons can be set, edited and manipulated by the user when selecting the "Context" menu button 104. For example, creation and/or management/editing of the user context can be preformed by a user with various options. One option is where the user selects specific entities and/or interactions to be inserted into the user context. Another option is to select local format objects (e.g., such as those describing entities and/or interactions) to be inputted and entered in the user context. Still another option is to select all or a portion of an existing biological diagram, convert the entities and interactions in such selection and enter the local format objects resulting from such conversion into the user context. Similarly, a user may create his/her own biological diagram, by freehand sketching or otherwise, convert it to local format objects and use these local format objects in the user context.

[0056] Simple rules are applied to break down the text into sentences. The identified nouns are represented in the "Entities" list 120, while the verbs are represented in the "Interactions" list 130, as noted above. A description of an example procedure employing simple rules follows. First the entire text is searched for the occurrence of a period, ".". Then each of these occurrences is

examined to throw away cases where the period character is not being used in the text to indicate the end of a sentence. An example of another use for this character is as a decimal point in a number.

[0057] For each sentence thus identified, the present invention searches that sentence for the presence of nouns specified in the user context. The present invention is able to recognize different grammatical forms of these nouns, such as plurals, even if they are not explicitly given in a user context. For each noun thus identified, the present invention creates an entity object. All known aliases for a noun, as specified by the user context, are recognized by the present invention as well. These aliases are all mapped to the appropriate single entity object. The location of the noun or one of its aliases within the text is also stored within the entity object according to the present invention. Optionally, a type may be assigned to the entity object, if such information is available in the user context entry for the noun from which the entity was created. If no type is available in user context, the present invention may assign a default type such as 'unknown' or it may attempt to compute a type based upon information which may include metadata about the text source which is being analyzed, other words which occur in the current sentence, or root words of known entities which occur as substrings in the name of the present entity. As more sentences are considered by the current invention, entities are not duplicated, that is, a user context specified noun or an alias which occurs in two separate sentences will be mapped to the same single entity object by the present invention.

[0058] Additionally, the present invention breaks down each sentence into individual words. Each of these possible words is first looked up in a dictionary of common English words. This dictionary was created from the dictionary available in the UNIX operating system. The system also stems the words in the dictionary to map different grammatical forms of the same word into one. For example, stemming maps both "proteins" and "protein" to the same stem "protein". Words that are not present in this dictionary are processed further as potential entities. These words are then looked up in a biological naming database. An example of such a database is BNS (Biological Naming System, see U.S. Patent Application Serial No. 10/154,529). For each occurrence of a word in such a database, an entity object is created by the present invention.

Aliases of database words are recognized in the present invention as well and all known aliases for a single word are mapped to the appropriate single entity object.

[0059] For each sentence thus identified, the present invention also searches that sentence for the presence of verbs specified in the user context. The present invention is able to recognize different grammatical forms of these verbs, such as different tenses, even if these different forms are not explicitly given in a user context entry. For each verb thus identified, the present invention creates an interaction object. Additionally, the present invention assigns all other entities which occur in the same sentence as the present verb to unassigned roles in the current interaction. The location of the verb within the text is also stored within the interaction object according to the present invention. Optionally, a type may be assigned to the interaction object, if such information is available in the user context entry for the verb from which the interaction was created. If no type is available in user context, the present invention may assign a default type such as 'unknown' or it may attempt to compute a type based upon information which may include metadata about the text source which is being analyzed, other words which occur in the current sentence, or root words of known Interactions which occur as substrings in the name of the present interaction.

[0060] The identified nouns are represented in the "Entities" list 120, while the verbs are represented in the "Interactions" list 130, as noted above. The present invention may display any available information about each Entity or Interaction in the label or icon of its representation in these list-based viewers. For example, the name and known aliases for an entity may be displayed. For an interaction, the name of that interaction along with the names of the entities contained in the interaction may be displayed. Furthermore, a textual representation of the roles played by the entities in that interaction may also be displayed at the interaction's label in the list-based viewer. An example list of roles an entity can play in an interaction includes, but not limited to, affecter, affected, unassigned, unknown, and mediator. In the current system, we have used the affecter, affected, and unassigned roles. However, unknown and mediator roles can also be assigned when a user doesn't know the actual role an entity plays or if the entity plays a mediator role and not the affecter or affected roles, respectively.

An example of displaying the roles of entities in an interaction includes, unassigned entities may be displayed in parentheses, while affecter and affected entities may be displayed on the left hand side and right hand side of an arrow, respectively. The present invention may also optionally use a coloring scheme to represent the labels for the entries in these list-based viewers. For example, each unique type of entity or interaction can be assigned to a particular color. The text for the labels of the entities and interactions displayed in these list-based viewers may be rendered in the color corresponding to the type of that entity or interaction. Further, the system allows the user to edit the aliases and names of the identified entities and interactions if some of the information is incorrect. The user can click the right mouse button and a list of action options is shown to the user. One of the options is an "Edit" option, that allows the user to edit the object (entity or interaction). Thus, any errors made by the automatic alias management routines may be manually corrected by the user.

[0061] Each entity and interaction in the "Entities" and "Interactions" panels, 120,130 respectively, also points back to all sections of text 110 that it occurs in, as provided for by the local format linking. For example, Fig. 2 shows an underlined sentence in the "Text" panel 110 with highlighted entities 112 and interactions 114 corresponding to the highlighted interaction 114 in the "Interactions" panel 130. Each interaction in the interaction panel 130 further lists the potential entities involved in the interaction along-with properties of the interaction and entities, context under which the interaction occurs, and any required conditions that have to be met. Each interaction also has slots or blank entry spaces provided where the user can assign entities to the particular roles that they play in that interaction. For example, an entity may be assigned to an "affecter" or "affected" role, depending if the entity is the subject or the object of the verb corresponding to the interaction of interest. This assignment of roles can be used for assigning directionality to the interaction, as it is consistent with current notation where an arrow is drawn to represent an interaction between two entities, to originate from the "affecter" entity and point toward and terminate at the "affected" entity. According to the present invention, in order to make such an assignment, a user can simply select, or click on the "+" icon to the left of the interaction to be worked on, which opens a list of subentries beneath the listed

interaction. An example of the results of such a selection by a user is shown in Fig. 3, where the "+" icon adjacent to the "increased" interaction 114 has been selected, which converts it to a "-" icon when the subentries menu opens. The user can then drag entities from the "Unassigned" subentry 132 to the "Affecter" subentry 134 or the "Affected" subentry 136. Entities 112 can also be dragged from the "Entities" list 120 into these subentries. Once entities are dragged into the respective subentries 134,136, the list title defines the relationship between entities (i.e., interaction) in an unambiguous way through this process of disambiguity, since the directionality of the interaction is defined by assignment of entities to the "affecter" and "affected" roles.

[0062] For example, completion of the disambiguation process for the entities and interactions highlighted in Fig. 2 would result in subentries under the interaction "increased" 114 listing HIP-55 under the Affecters subentry 134, and listing HPK1 as well as JNK1 under the Affected subentry 136. This gives the user a concise summary of the interaction. Any unassigned entities in the interaction are listed under the Unassigned subentry 132, and may be displayed in parentheses.

[0063] Tool 100 also allows the user to define new entities and interactions in these panels. This may be accomplished by pressing the right mouse button, for example, in either panel, which causes a pop-up menu to be presented to the user. Among the available options on the pop-up menu are "New Entity" and "New Interaction". Selection of one of these options causes a new editor window to appear. The new editor window may be used as an interface to create a new entity or interaction and associate the new entity or interaction with section(s) of the text, thereby mimicking the behavior of the automated analysis algorithm of the tools' software.

[0064] Optionally, tool 100 may include a diagramming window 150 which a user can drag interactions and entities into to display the interactions and entities diagrammatically. An example of a diagramming tool into which such a list may be dragged is described in co-pending, commonly assigned Application Serial No. 10/155,405, filed May 22, 2002 and titled "Database Model, Tools and Methods for Organizing Information Across External Information Objects". Application Serial No. 10/155,405 is hereby incorporated herein, in its entirety,

by reference thereto. Alternatively, the user may drag entities, prior to assigning them as affectors or affected in the Interactions window, directly into the diagramming tool 150. A determination as to whether an entity is an affecter or affected is determined, in this method, by the location that the entity is dragged to in the diagram. Thus, for example, if HIP-55 were dragged from the Entities list 120 to box 164, the results of which are shown at 170 in box 164 (Fig. 3), then an assignment of HIP-55 to the "affecter" role would automatically be made, as shown in the "Affecters" subentry 134. Since all the windows are simultaneous views of the same collection of local format objects, this assignment by the user automatically populates HIP-55 as an affector in the Interactions window 130 and in window 152, for example.

[0065] Accordingly, tool 100 provides functionality for the user to assign directionality and causality to interactions among entities. For potentially more powerful, natural and/or intuitive ways to display such information and acquire a user's assignments, the present invention further provides tools and methods for displaying a network of the relationships discussed above. It is generally most intuitive to display and manipulate networked information diagrammatically. For example, diagrams are used as a natural way of representing biochemical reactions or signaling pathways. Hence, Fig. 3 shows an example of the present invention provided by tool 100, which includes an optional graphical pane 150, as alluded to above. Pane 150 is adapted to contain a diagrammatic representation for one or more interactions. This feature provides the user with an alternate, intuitive mechanism to disambiguate and assign directionality and causality to extracted knowledge, resulting in a richer and more correct encoding of extracted biological knowledge.

[0066] The graphical pane 150 may be a single pane in which a simple graphical representation of an interaction is populated, or, alternatively may be divided into two or more areas, as shown in Fig. 3. When a single pane is used, entities may be dragged directly into the boxes 164 and 166 of an empty diagram to populate it with those entities so as to disambiguate the directionality of the interaction that the entities are involved in. In a multi-pane configuration, as shown, the left side pane may be a canvas area 152 which may be used for representing unassigned interactions and entities (e.g., see the boxes 164, 166

and arrow 162) and making assignments to them. On the right side palette areas 154,156 may be provided into which entities and interactions may be populated (such as by dragging and dropping, for example) 154. Further optionally, pane 156 may contain "building blocks" that can be used to construct custom graphical representations of biological interactions.

[0067] The initial setup for using tool 100 with the optional graphics functionality is the same as that described above for use of tool 100 without the graphics window. As a practical example, a user, such as a scientific researcher, may perform a scientific literature search to look for particular entities and/or interactions that have been identified over the course of some experimentation done. A scientific literature search may be performed, for example, using the tools and methods described in co-pending, commonly owned Application Serial No.10/033,823, filed Dec 19, 2001 and titled "Domain-Specific Knowledge-Based MetaSearch System and Methods of Using". Application Serial No. 10/033,823 is hereby incorporated herein, in its entirety, by reference thereto. The search results delivered may include a large number of textual documents that have been determined to be relevant to the search, which may define entities and/or interactions of interest.

[0068] Using viewer tool 100, an article, abstract thereof, or other selected portion thereof can be imported into window 110. Based upon user defined preferences, also known as the user context, (e.g., listing entities and interactions of interest, which is generally much more extensive than the search string that was used to perform the literature search), viewer 100, upon selecting or clicking on "Analyze Text" 102 automatically identifies the entities and interactions defined in the user defined preferences and highlights the same as shown in window 110. At the same time, the entities which are identified are populated in Entities window 120 and the interactions that are identified are populated in Interactions window 130.

[0069] Thus, when a textual document, such as a publication or an extract from a publication is inputted into the text viewer window 110 of tool 100, in a manner as described above, and the user selects the "Analyze Text" button 102, tool 100 responds by using user context and local format tools and methods to identify the nouns and verbs contained within the textual material, extract them,

and list them in the entities window 120 and interactions window 130, as shown. Each item in the list structure for interactions 130 is a visual representation of the underlying local format, restricted grammar representation of that interaction. Automated analysis using lexicons for entities and interactions may be used to identify the interesting nouns and verbs (e.g., those matching or related to an item in a lexicon) in every sentence of the text. Simple rules may be applied to break down the text into sentences. The identified nouns are represented in the "Entities" list 120, while the verbs are represented in the "Interactions" list 130, as noted above.

[0070] Each entity and interaction in the "Entities" and "Interactions" panels, 120,130 respectively, also points back to the part of text 110 it occurs in, as provided for by the local format linking, the same as was described above with regard to the example shown in Fig. 2. Each interaction in the interaction panel 130 further lists the potential entities involved in the interaction, along with properties of the interaction and entities, context under which the interaction occurs, and any required conditions that have to be met.

[0071] The user may next select one of the interactions from the interactions pane 130, such as by clicking on it with the mouse, for example, or through use of keyboard strokes. The canvas 152 and palette 154 may be, in response to the selection, automatically populated according to the contents of the highlighted interaction, as shown in Fig. 3. Canvas 152 may be populated with a diagram or graphical rendering 160 that is predesigned for a simple promotion interaction (i.e., describing an "increased" interaction), for example. The arrowed line 162 (which may be color-coded, e.g., red) running between the rectangles 164 and 166 indicates the promotion interaction and its direction, i.e., that entity 164 increases entity 166. The rectangles 166 (which may also be color coded, e.g., lavender) are "bounding boxes" for the affecter (box 164) and affected (box 166) entities, respectively. The entities involved in the interaction are shown in the palette 154. Icons for additional interactions and their directionality may be provided, as shown in palette 156, for example.

[0072] The user next begins to populate diagram 160 with entities from palette 154 by repetitively dragging entities from the palette area 154 over to one of the "bounding boxes" in the graphical pane. In this example, entity 170 (i.e., HIP-

55: HIP-55) is dragged from palette area 154 to the affecter box 164 and dropped. The result of this operation is shown in Fig. 3. The dragged/dropped entity 170 is then assigned by the system as an affecter 134 in the interaction "increased" 114 as also shown in Fig. 3. The user in this example next selects additional entities 172,174 from palette 154 area and drags them into the affected bounding box 166 and drops them there. Entities 172 and 174 are consequently automatically assigned as affecteds 136 in the interaction "increased" 114 as shown sequentially in Figs. 4A and 4B. Additionally or alternatively, entities and/or interactions may be dragged from the Entities panel 120 and/or Interactions panel 130 into the slots in the canvas area 152.

[0073] As a result of these actions, the user has graphically disambiguated the interaction and has automatically established this disambiguous relationship textually, resulting in both textual and graphical representations of the interaction which are directionally unambiguous. It is noted that that present invention may also be used to disambiguate according to the techniques described with reference to Fig. 2, to establish directionality in the local format text annotation in pane 130. The user may then select the "increased" interaction, for example, to automatically construct a filled in diagram, which would appear the same as the completed diagram 160 in Fig. 4B. By this functionality, the present invention serves as a tool for rapid and unambiguous construction of biological diagrams.

[0074] After disambiguating all of the knowledge extracted from a textual document in any of the manners described above, viewer 100 may next construct or draw a biological diagram representative of all of the disambiguated interactions that the document contains. Such construction may be done in an additional viewer pane, like the type shown in Fig. 5, for example, or sent to a diagram viewer 200. Of course, it is not necessary to complete disambiguation of all entities and interactions with regard to a textual document being examined, although this is the normal course that a user would follow. Diagram construction may be performed at any stage along the disambiguation process, at which time a diagram is constructed based upon entities and interactions which have been disambiguated thus far.

[0075] For example, a user may select an abstract, textual document, or a

portion thereof and import it into textual viewer 100 as described above. By engaging the "Analyze Text" 102 feature, the present invention identifies all entities and interactions in the textual document (or portion of a document) based on a predefined user context. The user context includes, for example, a list of keywords. Currently the present system is adapted to read XML or Excel files, although it would be apparent to one of ordinary skill in the art to extend the capability to other known formats. Each entry in the user context generally includes an identifier as to whether the entry is a noun or a verb; the name of the entry (i.e., which contributes to the lexicon for searching); the type that the entry is (e.g., cell, process, disease, or the like for nouns; bind, promote, inhibit, or the like for verbs); and aliases for the name of the entry, which are also added to the lexicon. However, the user context may still function with only a subset of such information, although less effectively (e.g., aliases could be omitted for some entries). Of course further descriptive information categories could be included for characterizing one or more entries in the user context, as would be readily apparent to one of ordinary skill in the art.

[0076] Additionally or alternatively, an existing diagram (whether manually drawn or a pre-existing machine format diagram) or portion thereof may be used to define a user context. Using a diagram viewer, such as described Application Serial No. (Application Serial No. not yet assigned, Attorney's Docket No. 10030687-1) for example, the diagram or portion thereof is converted to the local format. Once the conversion has been completed, the local format representation of the nouns and verbs represented diagrammatically are populated into the user context upon which a textual analysis may be based. More generally, any information which has been converted to the local format (e.g., experimental data, or other data) may be used to populate the user context.

[0077] If the user highlight or selects one or more entities and/or interactions from the entities and/or interactions lists 120,130, the tool automatically highlights those same entities and/or interactions in the text in window 110, via the linking provided by the local format data objects underlying the system. Because it is difficult to determine directionality of relationships/interactions identified solely through the use of natural language programming techniques, the user is involved, in the next step, in the process of disambiguating the

relationships, as noted above. Once interactions have been disambiguated, the user can select all or a portion of the disambiguated relationships listed, and a diagram view of the relationships is generated by linking like entities, using the local format architecture, for example. It should also be noted here, that a diagrammatic view can be generated prior to disambiguating all information and the disambiguation process can be performed on the diagram view, using techniques described above, as well as in Application Serial No. (Application Serial No. not yet assigned, Attorney's Docket No. 10030687-1).

[0078] The user may then wish to import another abstract, textual document, or a portion thereof, and iterate the process described above. After disambiguation of interactions/relationships in this next document, relationships and entities which are common to those identified in previous textual documents can be identified, either automatically or manually, and this information is joined to the previously created biological diagram.

[0079] Disambiguation can be performed with respect to each document, as the documents may not always agree as to the mechanism of an interaction. Where there is disagreement, upon generating a graphical diagram, the diagram will indicate such discord. As a simple example, if a first document indicates that entity A increases entity B, while a second article indicates that entity A decreases entity B, then the graphical representation may show a block for A, a block for B, and two lines extending between A and B. The lines may be differentiated by color coding (e.g., one green and one red) and/or by different arrowheads pointing towards B, e.g., one with an arrow-shaped arrowhead (i.e., -→) for a promotion and one with a blocked arrowhead (i.e., --- ı) to indicate an inhibition, for example. Other visual differentiators may be used in addition to, or alternatively to those described.

[0080] When more than one document contains an interaction that is displayed in a graphical diagram, annotations are made to that portion of the diagram which link that interaction to each of the documents where it occurs. Thus, not only is that interaction linked directly, such a by a hyperlink, but other annotations may be included to suit the user's needs, such as described for example in Application Serial Nos. 10/155,304, 10/155,616 and 09/863,115, all of which are incorporated herein, in their entireties, by reference thereto.

[0081] Thus, the present invention further provides the ability to build networks of interactions by composing entities, interactions, and diagrams. Using this feature, the user selects a subset of interactions in the "Interactions" list 130 and drags them into a separate network viewer window 190, as shown in Fig. 5. Alternatively, the selected interactions may be automatically populated into viewer window 190 upon their selection. The system may merge interactions with common entities, forming a graph structure.

[0082] The graph structure can be built upon by analyzing an additional textual document and processing it as described above with regard to the first textual document. Upon identifying and disambiguating the interactions in the second textual document, these interactions can then be joined in the graphical composition. This type of building can be done repeatedly with as many textual documents as desired.

[0083] The present invention further provides the ability to compare extracted knowledge against an existing biological network. The user can load an existing network diagram into the system or select a subset of an existing network via search. The system overlays the extracted interactions and their entities upon the imported diagram, such as by color-coding those nodes and arcs in the imported diagram that correspond to extracted entities and interactions, for example. An example of this functionality, using a Map-kinase signaling pathway diagram 210 imported from the KEGG web site (http://www.kegg.org), is schematically shown in Figs. 6A-6B. The color-coded nodes 212 and links 214 indicate the overlays of the extracted interactions and their entities. A further aspect of this overlay technique uses an automated approach to search for existing networks that contain a user-specified set of interactions. The networks found to include the specified set of interactions are then provided to the user for selection among this set to overlay the extracted interactions and entities.

[0084] The present invention may also be deployed in batch mode, in conjunction with a biological textual search tool, such as that described in co-pending, commonly owned Application Serial No.10/033,823, for example. In this mode, the researcher may perform a scientific literature search, as described above, but instead of simply analyzing one document at a time, tool 100 may be tasked to analyze all documents (or some subset thereof) returned from the

search, in batch mode. The result is a combined list of entities and interactions extracted from all of the documents processed. This may simplify the user's ability to compare and contrast among the textual documents during the process of disambiguation which can be performed next. Also, any generation of larger scale diagrams from the disambiguated entities and interactions may be facilitated, by the user being able to view all entities and interactions together.

[0085] Upon selecting an interaction in batch mode, text viewer window 110 may be adapted to identify each textual document that the interaction occurs in. For an example of batch processing, a literature search may be performed using a domain-specific, knowledge-based metasearch system, such as that described in Application Serial No. 10/033,823, which allows the user to specify particular scientific databases to search. The search may return a sizeable number of relevant documents. For example, assuming fifty to one hundred relevant documents are returned in the search results, then the system described in Application Serial No. 10/033, 823 may be used to extract relevant text from these documents in batch mode, e.g., to text mine and identify abstracts and portions of text containing the search words.

[0086] Once such portions and abstracts have been identified and imported into tool 100, the user may select "Analyze Text" and identify all nouns and interactions in the entire batch of identified portions and abstracts, based on a predefined user context. By highlighting any identified noun or interaction, an annotation link shows, in the text viewer, where in text the highlighted entity(ies) and/or interaction(s) is/are located. It is difficult to determine directionality of relationships/interactions using natural language programming alone. Therefore, the user is involved, in the next step, in the process of disambiguating the relationships, just as described above with regard to non-batch processing modes. Once all interactions have been disambiguated, the user can select all or a portion of the relationships listed, and a diagram view of the relationships is generated by linking like entities.

[0087] Disambiguation can be performed with respect to each document, as well as across a corpus of documents, by processing in batch mode. However, all documents may not always agree as to the mechanism of an interaction. Where there is disagreement, upon generating a graphical diagram, the diagram will

indicate such discord. As a simple example, if a first document indicates that entity A increases entity B, while a second article indicates that entity A decreases entity B, then the graphical representation may show a block for A, a block for B, and two lines extending between A and B. The lines may be differentiated by color coding (e.g., one green and one red) and/or by different arrowheads (e.g., an arrow-shaped arrowhead indicating promotion, and a blocked arrowhead indicating inhibition) each pointing toward block B. Other visual differentiators may be used in addition to, or alternatively to those described.

[0088] When more than one document contains an interaction that is displayed in a graphical diagram, annotations are made to that portion of the diagram which link that interaction to each of the documents where it occurs. Thus, not only is that interaction linked directly, such a by a hyperlink, but other annotations may be included to suit the user's needs.

[0089] The present invention may also be used for alias management, that is to track and equate various names that are used for the same entity or interaction, for example, as described above. This function may be used to supplement the Biological Information Naming System described in co-pending, commonly owned Application Serial No. 10/154,529, filed May 22, 2002, and titled "Biotechnology Information Naming System", which is incorporated herein, in its entirety, by reference thereto. For example, referring back to Fig. 2, tool 100 identified two entities 118 (JNK) and 119 (JNK1) among all of the entities identified. Employing the Biotechnology Information Naming System to look up each entity, tool 100 then determines that JNK and JNK1 refer to the same entity (the formal name of which is MAPK8). Upon making this determination, both JNK and JNK1 are automatically listed on the same entity line in Entities window 120, as shown in Fig. 2. Additionally, tool 100 manages these aliases by highlighting both in the text viewer 110 when the user is interested in identifying interactions involving either JNK or JNK1. This provides a powerful visual reminder to the user of the equivalency among the aliases.

[0090] Fig. 7 illustrates a typical computer system in accordance with an embodiment of the present invention. The computer system 800 includes any number of processors 802 (also referred to as central processing units, or CPUs)

that are coupled to storage devices including primary storage 806 (typically a random access memory, or RAM), primary storage 804 (typically a read only memory, or ROM). As is well known in the art, primary storage 804 acts to transfer data and instructions uni-directionally to the CPU and primary storage 806 is used typically to transfer data and instructions in a bi-directional manner Both of these primary storage devices may include any suitable computer-readable media such as those described above. A mass storage device 808 is also coupled bi-directionally to CPU 802 and provides additional data storage capacity and may include any of the computer-readable media described above. Mass storage device 808 may be used to store programs, data and the like and is typically a secondary storage medium such as a hard disk that is slower than primary storage. It will be appreciated that the information retained within the mass storage device 808, may, in appropriate cases, be incorporated in standard fashion as part of primary storage 806 as virtual memory. A specific mass storage device such as a CD-ROM 814 may also pass data uni-directionally to the CPU.

[0091] CPU 802 is also coupled to an interface 810 that includes one or more input/output devices such as such as video monitors, track balls, mice, keyboards, microphones, touch-sensitive displays, transducer card readers, magnetic or paper tape readers, tablets, styluses, voice or handwriting recognizers, or other well-known input devices such as, of course, other computers. Finally, CPU 802 optionally may be coupled to a computer or telecommunications network using a network connection as shown generally at 812. With such a network connection, it is contemplated that the CPU might receive information from the network, or might output information to the network in the course of performing the above-described method steps. The above-described devices and materials will be familiar to those of skill in the computer hardware and software arts.

[0092] The hardware elements described above may implement the instructions of multiple software modules for performing the operations of this invention. For example, instructions for text mining and conversion to the local format may be stored on mass storage device 808 or 814 and executed on CPU 808 in conjunction with primary memory 806.

[0093] In addition, embodiments of the present invention further relate to computer readable media or computer program products that include program instructions and/or data (including data structures) for performing various computer-implemented operations. The media and program instructions may be those specially designed and constructed for the purposes of the present invention, or they may be of the kind well known and available to those having skill in the computer software arts. Examples of computer-readable media include, but are not limited to, magnetic media such as hard disks, floppy disks, and magnetic tape; optical media such as CD-ROM, CDRW, DVD-ROM, or DVD-RW disks; magneto-optical media such as floptical disks; and hardware devices that are specially configured to store and perform program instructions, such as read-only memory devices (ROM) and random access memory (RAM). Examples of program instructions include both machine code, such as produced by a compiler, and files containing higher level code that may be executed by the computer using an interpreter.

[0094] While the present invention has been described with reference to the specific embodiments thereof, it should be understood by those skilled in the art that various changes may be made and equivalents may be substituted without departing from the true spirit and scope of the invention. In addition, many modifications may be made to adapt a particular model, tool, process, process step or steps, to the objective, spirit and scope of the present invention. All such modifications are intended to be within the scope of the claims appended hereto.